PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-217720
(43)Date of publication of application : 10.08.2001

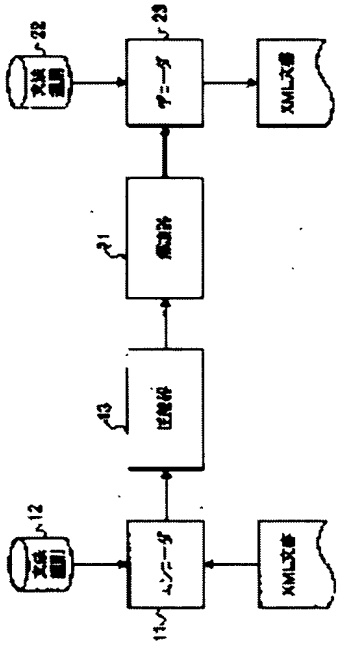
(51)Int.Cl. H03M 7/30

(21)Application number : 2000-028359
(22)Date of filing : 04.02.2000

(71)Applicant : INTERNATL BUSINESS MACH CORP <IBM>
(72)Inventor : MARUYAMA HIROSHI
TAMURA TAKETO
URAMOTO NAOHIKO

(54) DATA COMPRESSING APPARATUS, DATA BASE SYSTEM, DATA COMMUNICATION SYSTEM, DATA COMPRESSING METHOD, STORAGE MEDIUM AND PROGRAM TRANSMITTER

(57)Abstract:
PROBLEM TO BE SOLVED: To provide a data compression for encoding the structural part of a document in a tree local language such as XML or ASN.1.
SOLUTION: In the data compressing apparatus for encoding and compressing data, there are provided a grammar rule 12 for the tree local language with which the data are expressed in a labeled tree structure, an encoder 11 for reading a document described in this tree local language, separating this document into a structure and contents and encoding this structure while using the grammar rule 12 and a compressor 13 for compression-encoding the contents of this document extracted by the encoder 11.



LEGAL STATUS

[Date of request for examination] 12.09.2000
[Date of sending the examiner's decision of rejection]
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]
[Patent number] 3368883

[Date of registration] 15.11.2002
[Number of appeal against examiner's decision of rejection]
[Date of requesting appeal against examiner's decision of rejection]
[Date of extinction of right]

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号
特開2001－217720
（P2001－217720A）
(43)公開日　平成13年８月10日(2001.8.10)

(51)Int.Cl.⁷	識別記号	ＦＩ	ﾃｰﾏｺｰﾄﾞ(参考)
Ｈ０３Ｍ　7/30		Ｈ０３Ｍ　7/30	Ｚ　５Ｊ０６４

審査請求　有　　請求項の数13　ＯＬ　（全 12 頁）

(21)出願番号　特願2000－28359(P2000－28359)

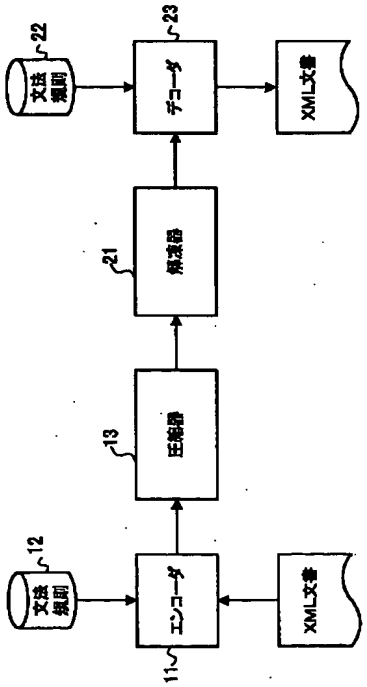
(22)出願日　平成12年２月４日(2000.2.4)

(71)出願人　390009531
インターナショナル・ビジネス・マシーンズ・コーポレーション
ＩＮＴＥＲＮＡＴＩＯＮＡＬ　ＢＵＳＩＮＥＳＳ　ＭＡＳＣＨＩＮＥＳ　ＣＯＲＰＯＲＡＴＩＯＮ
アメリカ合衆国10504、ニューヨーク州　アーモンク　（番地なし）
(74)代理人　100086243
弁理士　坂口　博　（外３名）

最終頁に続く

(54)【発明の名称】　データ圧縮装置、データベースシステム、データ通信システム、データ圧縮方法、記憶媒体及びプログラム伝送装置

(57)【要約】
【課題】　ＸＭＬやＡＳＮ．１等の木ローカル言語における文書の構造部分の符号化を行うデータ圧縮を実現する。
【解決手段】　データを符号化して圧縮するデータ圧縮装置であって、データがラベル付きの木構造で表現される木ローカル言語の文法規則１２と、この木ローカル言語で記述された文書を読み込んで、この文書を構造とコンテンツとに分け、文法規則１２を用いてこの構造を符号化するエンコーダ１１と、このエンコーダ１１によって抽出されたこの文書のコンテンツを圧縮符号化する圧縮器１３とを備える。



1

【特許請求の範囲】

【請求項１】　データを符号化して圧縮するデータ圧縮装置であって、

データがラベル付きの木構造で表現される木ローカル言語の文法規則を格納した文法格納部と、

前記木ローカル言語で記述された文書を読み込んで、当該文書を構造とコンテンツとに分け、前記文法格納部に格納された前記文法規則を用いて当該構造を符号化するエンコーダと、

前記エンコーダによって抽出された前記文書のコンテンツを圧縮符号化する圧縮器とを備えることを特徴とするデータ圧縮装置。

【請求項２】　前記エンコーダは、

処理対象である文書を構造とコンテンツとに分ける分割処理部と、

前記文法規則に対応するプッシュダウンオートマトンを構築するオートマトン構築部と、

前記オートマトン構築部により構築された前記プッシュダウンオートマトンを用いて、前記分割処理部により分割された前記文書の構造に対する構文解析を行い、当該構造の符号化されたデータ列を生成する符号化データ生成部とを備えることを特徴とする請求項１に記載のデータ圧縮装置。

【請求項３】　前記エンコーダの前記符号化データ生成部は、前記オートマトン構築部により構築された前記プッシュダウンオートマトン中に存在する選択肢に対して符号を割り当て、当該プッシュダウンオートマトンを走らせて前記文書の構造を解析した際に当該選択肢の箇所で選択された選択肢に割り当てられている当該符号を出力することにより、当該構造の符号化データ列を生成することを特徴とする請求項２に記載のデータ圧縮装置。

【請求項４】　前記圧縮器は、前記文書のコンテンツと共に、前記エンコーダにて符号化された当該文書の構造に対しても圧縮符号化を行うことを特徴とする請求項１に記載のデータ圧縮装置。

【請求項５】　通信ネットワークを介してデータ送信を行う送信側データ処理装置と、当該送信側データ処理装置から送信されたデータを当該通信ネットワークを介して受信する受信側データ処理装置とを備えたデータ通信システムであって、

前記送信側データ処理装置は、

データがラベル付きの木構造で表現される木ローカル言語の文法規則を格納した第１の文法格納部と、

前記木ローカル言語で記述された送信文書を読み込んで、当該送信文書を構造とコンテンツとに分け、前記第１の文法格納部に格納された前記文法規則を用いて当該構造を符号化するエンコーダと、

前記エンコーダによって抽出された前記送信文書のコンテンツを圧縮符号化する圧縮器と、

前記エンコーダにより符号化された前記構造及び前記圧縮器により圧縮符号化された前記コンテンツを送信する送信部とを備え、

2

前記受信側データ処理装置は、

前記送信側データ処理装置から送信された受信する受信部と、

前記送信側データ処理装置の前記第１の文法格納部に格納された文法規則と同一内容の文法規則を格納した第２の文法格納部と、

前記送信側データ処理装置の前記圧縮器による圧縮符号化手法に対応する解凍手法にて、前記受信部が受信した受信データのうち前記送信文書のコンテンツに対応するデータを解凍する解凍器と、

前記受信部が受信した受信データのうち前記送信文書の構造に対応するデータを、前記第２の文法格納部に格納された前記文法規則を用いて当該構造を復号化するデコーダとを備えることを特徴とするデータ通信システム。

【請求項６】　データを記憶装置に格納して管理するデータベースシステムであって、

データがラベル付きの木構造で表現される木ローカル言語の文法規則を格納した文法格納部と、

前記木ローカル言語で記述された文書を読み込んで、当該文書を構造とコンテンツとに分け、前記文法格納部に格納された前記文法規則を用いて当該構造を符号化するエンコーダと、

前記エンコーダによって抽出された前記文書のコンテンツを圧縮符号化する圧縮器と、

前記エンコーダにより符号化された前記文書の構造と前記圧縮器により圧縮符号化された前記文書のコンテンツとを格納する記憶装置とを備えることを特徴とするデータベースシステム。

【請求項７】　前記圧縮器は、前記文書のコンテンツと共に、前記エンコーダにて符号化された当該文書の構造に対しても圧縮符号化を行うことを特徴とする請求項６に記載のデータベースシステム。

【請求項８】　データを符号化して圧縮するデータ圧縮方法であって、

データがラベル付きの木構造で表現される木ローカル言語で記述された文書を読み込んで、当該文書を構造とコンテンツとに分けるステップと、

前記木ローカル言語の文法規則を用いて、前記文書の構造を符号化するステップと、

前記文書のコンテンツを圧縮符号化するステップとを含むことを特徴とするデータ圧縮方法。

【請求項９】　前記文書の構造を符号化するステップは、

前記文法規則に対応するプッシュダウンオートマトンを構築するステップと、

前記プッシュダウンオートマトン中に存在する選択肢に対して符号を割り当てるステップと、

前記プッシュダウンオートマトンを走らせて前記文書の

構造を深さ優先でたどりながら解析し、当該選択肢の箇所で選択された選択肢に割り当てられている前記符号を出力するステップと、

前記プッシュダウンオートマトンを走らせて出力された前記符号の列を前記文書の構造の符号化されたデータ列として出力するステップとを含むことを特徴とする請求項８に記載のデータ圧縮方法。

【請求項１０】　前記文書の構造を符号化するステップに先だって、

処理対象である前記木ローカル言語の文書のノードに属性が含まれている場合に、当該属性を、当該属性を持つ要素の子ノードに変換することにより、前記木ローカル言語の文法規則及び前記文書を、前記プッシュダウンオートマトンで扱える木構造に変換するステップをさらに含むことを特徴とする請求項９に記載のデータ圧縮方法。

【請求項１１】　前記文書の構造を符号化するステップの後に、

汎用的な他の圧縮符号化手法を用いて、当該符号化された当該文書の構造をさらに圧縮符号化するステップをさらに含むことを特徴とする請求項８に記載のデータ圧縮方法。

【請求項１２】　コンピュータに実行させるプログラムを当該コンピュータの入力手段が読取可能に記憶した記憶媒体において、

前記プログラムは、

データがラベル付きの木構造で表現される木ローカル言語で記述された文書を読み込んで、当該文書を構造とコンテンツとに分ける処理と、

前記木ローカル言語の文法規則を用いて、前記文書の構造を符号化する処理と、

前記文書のコンテンツを圧縮符号化する処理とを前記コンピュータに実行させることを特徴とする記憶媒体。

【請求項１３】　コンピュータに、

データがラベル付きの木構造で表現される木ローカル言語で記述された文書を読み込んで、当該文書を構造とコンテンツとに分ける処理と、前記木ローカル言語の文法規則を用いて、前記文書の構造を符号化する処理と、前記文書のコンテンツを圧縮符号化する処理とを実行させるプログラムを記憶する記憶手段と、

前記記憶手段から前記プログラムを読み出して当該プログラムを送信する送信手段とを備えたことを特徴とするプログラム伝送装置。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は、ＸＭＬやＡＳＮ．１等の木ローカル言語で記述された文書を圧縮するデータ圧縮方法に関する。

【０００２】

【従来の技術】ＸＭＬ（Extensible Markup Language）

は、文書の意味構造を簡単なマークで記述(マークアップ)するマークアップ言語の一種である。ＸＭＬでは、文法を規定し、文書の構成要素に論理的意味を持たせることにより、ユーザが独自の拡張を行うことが可能である。そのため、インターネットにおけるデータ交換に用いるデータフォーマットとして期待されている。

【０００３】ＸＭＬには、ＤＴＤ（文書型定義：Document Type Definition）という概念があり、あるＤＴＤに関して、文書が妥当（valid）かどうかを判定できる。具体的には、例えば、<BOOK>というノードの下には、<TITLE>、<AUTHOR>、<PUBLISHER>というノードがこの順にそれぞれ１回ずつ現れる、というような文法規則を規定する。そして、所定の文書が妥当かどうか、すなわち、当該文法規則に則しているかどうかを判定することができる。

【０００４】ＸＭＬ文書の構造は、形式言語的には、木言語のうちの木ローカルと呼ばれるクラスに属する。木ローカル言語とは、データがラベル付きの木構造で表現され、かつ、正しいデータが木の各ノードラベルに対して子ノードのラベルの正規言語で規定されるような木言語である。すなわち、木ローカル言語では、所定の文法（ＸＭＬにおいてはＤＴＤにて規定される）に属する木の集合が、各ノードの子ノードのリストを規定する正規言語によって決定付けられる。この種の木ローカル言語としては、他に、ＡＳＮ．１（Abstract Syntax Notation 1）等がある。

【０００５】近年、ＸＭＬを用いて、ネットワーク上のリソースやＨＴＭＬでは実現不可能であった複雑なデータ構造を記述し、ビジネスアプリケーションを構築する動きが広がっている。このようなシステムでは、大規模なＸＭＬ文書がアプリケーション間で交換されることが予想される。一般に、データ交換を行ったりデータベースにデータを格納したりする際には、送信効率の向上や格納するデータファイルのサイズを縮小するために、データファイルの圧縮を行う。そのため、種々のデータフォーマットに対して適用できる汎用的なデータ圧縮技術や、特定のデータフォーマットに特化したデータ圧縮技術などが数多く提案されている。したがって、ＸＭＬ文書の交換においてもこれらのデータ圧縮技術を適用することが考えられる。なお、ＸＭＬ文書は、タグ名などにかなりの冗長度があるため、高い圧縮率が期待できるデータ構造であるといえる。

【０００６】

【発明が解決しようとする課題】上述したように、データ交換を行ったりデータベースにデータを格納したりする際には、データファイルの圧縮を行うことが一般的である。そして、ＸＭＬ等の木ローカル言語では、タグなどの文書構造を示す情報部分に対して高い圧縮率を期待できる。

【０００７】例えば、データ通信において、通信の当事

者が文法Ｇを共有しており、互いに当該文法Ｇに対して妥当なＸＭＬ文書のみをやりとりすることが保証される場合を考える。また、上記のように文法Ｇにおいて、<BOOK>というノードの下には、<TITLE>、<AUTHOR>、<PUBLISHER>というノードがこの順にそれぞれ１回ずつ現れる、という文法規則が規定されているものとする。この場合、ＸＭＬ文書の受信者は、受信したＸＭＬ文書の中で、例えば<BOOK>というタグを検出した場合に、その最初の子ノードは<TITLE>であると予測することができる。したがって、<TITLE>という情報は、この仮定のもとでは冗長である。この考えを用いることで、メッセージ長を最適化するような、ＸＭＬ文書のエンコーディング方法が可能である。このことは、ＸＭＬに限らず、任意の木ローカル言語（例えば、ＡＳＮ．１）に対しても同様である。しかしながら、従来、このような木ローカル言語における文書の構造部分を符号化する圧縮方法は何ら提案されていなかった。

【０００８】そこで本発明は、ＸＭＬやＡＳＮ．１等の木ローカル言語における文書の構造部分の符号化を行うデータ圧縮を実現する。

【０００９】また本発明は、木ローカル言語に特化すると共に、他の汎用的なデータ圧縮技術と併用することにより、高い圧縮率を期待できるデータ圧縮方法を提供する。

【００１０】

【課題を解決するための手段】かかる目的のもと、本発明は、データを符号化して圧縮するデータ圧縮装置であって、データがラベル付きの木構造で表現される木ローカル言語の文法規則を格納した文法格納部と、この木ローカル言語で記述された文書を読み込んで、この文書を構造とコンテンツとに分け、文法格納部に格納された文法規則を用いてこの構造を符号化するエンコーダと、このエンコーダによって抽出されたこの文書のコンテンツを圧縮符号化する圧縮器とを備えることを特徴としている。木ローカル言語とは、データがラベル付きの木構造で表現され、かつ、正しいデータが木の各ノードラベルに対して子ノードのラベルの正規言語で規定されるような木言語である。

【００１１】ここで、このエンコーダは、処理対象である文書を構造とコンテンツとに分ける分割処理部と、文法規則に対応するプッシュダウンオートマトンを構築するオートマトン構築部と、このオートマトン構築部により構築された前記プッシュダウンオートマトンを用いて、前記分割処理部により分割された前記文書の構造に対する構文解析を行い、当該構造の符号化されたデータ列を生成する符号化データ生成部とを備えることを特徴としている。

【００１２】さらに、このエンコーダの符号化データ生成部は、オートマトン構築部により構築されたこのプッシュダウンオートマトン中に存在する選択肢に対して符号を割り当て、このプッシュダウンオートマトンを走らせて木ローカル言語で記述された文書の構造を解析した際にこの選択肢の箇所で選択された選択肢に割り当てられている符号を出力することにより、この構造の符号化データ列を生成することを特徴としている。このような構成とすれば、タグなどのラベルを用いて記述されたこの文書の構造を簡単な符号列に変換（符号化）することが可能となる。プッシュダウンオートマトンを用いて文書の構造を解析する際には、この文書の木構造を深さ優先でたどりながら解析を行う。すなわち親ノードからの距離が等しい階層ごとに解析していくのではなく、深さ方向のノードどうしのつながり（親子関係）を優先させて木をたどりながら解析する。

【００１３】またここで、圧縮器は、この木ローカル言語で記述された文書のコンテンツと共に、エンコーダにて符号化されたこの文書の構造に対しても圧縮符号化を行うことを特徴としている。この圧縮器による圧縮手法は、特に制限はなく、従来から用いられている汎用的な手法を採用することができる。エンコーダにより文書の構造を符号化した結果、ある程度規則的なデータ列が得られる場合は、この符号化データ列に対し、ＰＫＺＩＰ等の汎用的な手法を用いてさらに圧縮符号化を行うことにより、高い圧縮率を期待できる。そこで、文書のコンテンツを圧縮する際に、この符号化データ列を合わせて圧縮することが好ましい。さらに、同じ文書における構造の符号化データ列とコンテンツとを結合した上で圧縮を行うことにより、構造とコンテンツとが別ファイルとなることを避けられるので、ファイル管理の上でも好ましい。

【００１４】また、本発明は、通信ネットワークを介してデータ送信を行う送信側データ処理装置と、この送信側データ処理装置から送信されたデータをこの通信ネットワークを介して受信する受信側データ処理装置とを備えたデータ通信システムであって、この送信側データ処理装置は、データがラベル付きの木構造で表現される木ローカル言語の文法規則を格納した第１の文法格納部と、この木ローカル言語で記述された送信文書を読み込んで、この送信文書を構造とコンテンツとに分け、第１の文法格納部に格納された文法規則を用いてこの送信文書の構造を符号化するエンコーダと、このエンコーダによって抽出されたこの送信文書のコンテンツを圧縮符号化する圧縮器と、エンコーダにより符号化された構造及び圧縮器により圧縮符号化されたコンテンツを送信する送信部とを備え、この受信側データ処理装置は、この送信側データ処理装置から送信された受信する受信部と、送信側データ処理装置の第１の文法格納部に格納された文法規則と同一内容の文法規則を格納した第２の文法格納部と、送信側データ処理装置の圧縮器による圧縮符号化手法に対応する解凍手法にて、この受信部が受信した受信データのうち送信文書のコンテンツに対応するデー

タを解凍する解凍器と、この受信部が受信した受信データのうち送信文書の構造に対応するデータを、第２の文法格納部に格納された前記文法規則を用いて当該構造を復号化するデコーダとを備えることを特徴としている。このように、データの送信側と受信側とで予め共通の文法規則を用意しておけば、データ通信において、木ローカル言語で記述された文書に対するきわめて圧縮率の高い圧縮を行うことができ、通信効率を向上させることができる点できわめて優れている。なお、ビジネス間通信においては、木ローカル言語の文法規則として共通のものを用いることが予め定められることが一般的なので、本発明を導入することは容易である。

【００１５】さらにまた、本発明は、データを記憶装置に格納して管理するデータベースシステムであって、データがラベル付きの木構造で表現される木ローカル言語の文法規則を格納した文法格納部と、この木ローカル言語で記述された文書を読み込んで、この文書を構造とコンテンツとに分け、文法格納部に格納された文法規則を用いてこの文書の構造を符号化するエンコーダと、このエンコーダによって抽出された文書のコンテンツを圧縮符号化する圧縮器と、エンコーダにより符号化されたこの文書の構造と圧縮器により圧縮符号化されたこの文書のコンテンツとを格納する記憶装置とを備えることを特徴としている。

【００１６】ここで、この圧縮器は、前記文書のコンテンツと共に、前記エンコーダにて符号化された当該文書の構造に対しても圧縮符号化を行うことを特徴としている。同じ文書における構造の符号化データ列とコンテンツとを結合した上で圧縮を行うことにより、高い圧縮率を期待でき、さらに構造とコンテンツとが別ファイルとなることを避けられるので、ファイル管理の上でも好ましい。

【００１７】また、本発明は、データを符号化して圧縮するデータ圧縮方法であって、データがラベル付きの木構造で表現される木ローカル言語で記述された文書を読み込んで、この文書を構造とコンテンツとに分けるステップと、この木ローカル言語の文法規則を用いて、この文書の構造を符号化するステップと、この文書のコンテンツを圧縮符号化するステップとを含むことを特徴としている。

【００１８】ここで、文書の構造を符号化するステップは、文法規則に対応するプッシュダウンオートマトンを構築するステップと、プッシュダウンオートマトン中に存在する選択肢に対して符号を割り当てるステップと、このプッシュダウンオートマトンを走らせて前記文書の構造を深さ優先でたどりながら解析し、この選択肢の箇所で選択された選択肢に割り当てられているこの符号を出力するステップと、このプッシュダウンオートマトンを走らせて出力されたこの符号の列をこの木ローカル言語で記述された文書の構造の符号化されたデータ列として出力するステップとを含むことを特徴としている。このような構成とすれば、タグなどのラベルを用いて記述されたこの文書の構造を簡単な符号列に変換（符号化）することが可能となる。

【００１９】このデータ圧縮方法において、木ローカル言語で記述された文書の構造を符号化するステップに先だって、処理対象であるこの木ローカル言語の文書のノードに属性が含まれている場合に、この属性を、この属性を持つ要素の子ノードに変換することにより、この木ローカル言語の文法規則及び文書を、プッシュダウンオートマトンで扱える木構造に変換するステップをさらに含むことを特徴としている。このように構成すれば、ＸＭＬのように処理対象の文書に属性が含まれている場合にも、プッシュダウンオートマトンを用いた構造の符号化を行うことができる点で好ましい。

【００２０】さらに、文書の構造を符号化するステップの後に、汎用的な他の圧縮符号化手法を用いて、符号化された文書の構造をさらに圧縮符号化するステップをさらに含むことを特徴としている。このような構成とすることにより、さらに高い圧縮率を期待できる点で好ましい。

【００２１】また、本発明は、コンピュータに実行させるプログラムを当該コンピュータの入力手段が読取可能に記憶した記憶媒体において、このプログラムは、データがラベル付きの木構造で表現される木ローカル言語で記述された文書を読み込んで、この文書を構造とコンテンツとに分ける処理と、この木ローカル言語の文法規則を用いて、この文書の構造を符号化する処理と、この文書のコンテンツを圧縮符号化する処理とをこのコンピュータに実行させることを特徴としている。このような構成とすれば、このプログラムをインストールしたあらゆる情報処理装置において、この木ローカル言語にて記述された文書を高い圧縮率で圧縮することができ、通信効率や記憶効率を向上させることができる。

【００２２】さらにまた、本発明は、コンピュータに、データがラベル付きの木構造で表現される木ローカル言語で記述された文書を読み込んで、この文書を構造とコンテンツとに分ける処理と、この木ローカル言語の文法規則を用いて、この文書の構造を符号化する処理と、この文書のコンテンツを圧縮符号化する処理とを実行させるプログラムを記憶する記憶手段と、この記憶手段からこのプログラムを読み出してこのプログラムを送信する送信手段とを備えたことを特徴としている。このような構成とすれば、このプログラム伝送装置からこのプログラムをダウンロードしてインストールしたあらゆる情報処理装置において、この木ローカル言語にて記述された文書を高い圧縮率で圧縮することができ、通信効率や記憶効率を向上させることができる。

【００２３】

【発明の実施の形態】以下、添付図面に示す実施の形態

に基づいてこの発明を詳細に説明する。図１は、本実施の形態における文書圧縮システムの全体構成を説明する図である。図１において、符号１１はエンコーダであり、圧縮対象である文書を構造とコンテンツとに分け、その構造部分を、所定の記憶装置に記憶された文法規則１２を用いて符号化する。符号１３は圧縮器であり、エンコーダ１１により符号化された構造部分と、当該文書のコンテンツ部分とを圧縮する。符号２１は解凍器であり、圧縮器１３により圧縮された文書を解凍する。解凍器２１により解凍された時点では、当該文書は、コンテンツ部分とエンコーダ１１により符号化された構造部分とに分かれている。符号２３はデコーダであり、符号化されている構造部分を、所定の記憶装置に記憶された文法規則２２を用いて複合化し、コンテンツ部分と合わせて、文書を復元する。

【００２４】本実施の形態をデータ通信の際のデータ圧縮に用いる場合は、エンコーダ１１及び圧縮器１３は送信側に置かれ、解凍器２１及びデコーダ２３は受信側に置かれることとなる。また、データベースシステムにおいて格納するデータファイルを圧縮するために用いる場合は、データの流れに応じて、エンコーダ１１がデコーダ２３として動作し、圧縮器１３が解凍器２１として動作する。

【００２５】以下、処理対象の木ローカル言語としてＸＭＬを用いる場合を例として説明する。図２は、本実施の形態によるデータ圧縮の手順を説明する図である。図２に示す本実施の形態によるデータ圧縮処理では、まず、処理対象であるＸＭＬ文書２０１が、エンコーダ１１に読み込まれて、構造２０２とコンテンツ２０４とに分解される。ここで、構造２０２とは、当該ＸＭＬ文書２０１の木構造、タグ名及び属性名であり、コンテンツ２０４とは、当該ＸＭＬ文書２０１の＃ＰＣＤＡＴＡ及び属性値である。ＸＭＬ文書２０１を構造２０２とコンテンツ２０４とに分解するのは、構造２０２とコンテンツ２０４とは、通常全く異なる統計的偏りを持っているため、独立して圧縮した方が効果的だからである。

【００２６】次に、ＸＭＬ文書２０１を分解して得られた構造２０２が、エンコーダ１１により符号化される。構造２０２の符号化には、文法規則１２が用いられる。ここでは、処理対象をＸＭＬ文書としているので、文法規則１２はＤＴＤにより規定される。符号化処理の詳細な内容については後述する。符号化された結果である符号化データ列２０３とコンテンツ２０４とは、圧縮器１３に送られる。

【００２７】最後に、圧縮器１３において、符号化データ列２０３とコンテンツ２０４とが圧縮符号化され、これらのデータを合わせて、圧縮されたＸＭＬ文書２０５が生成される。圧縮器１３による符号化処理には、ＬＺ７７等の既知の適当な手法を用いる。ここで、圧縮器１３による圧縮符号化は、主としてコンテンツ２０４に対して行われることとなる。しかしながら、ＰＫＺＩＰ等の汎用的な圧縮符号化方式は、符号化データ列２０３に対しても有効である。後述するように、本実施例では、符号化データ列２０３は数字列として出力される。したがって、ある程度規則的な数字列であるような場合は、高い圧縮率が期待できる。そこで、圧縮器１３では、コンテンツ２０４と共に符号化データ列２０３も圧縮符号化を行う。なお、符号化データ列２０３に対して圧縮器１３による圧縮を行うか否かは任意である。符号化データ列２０３と圧縮器１３により圧縮されたコンテンツ２０４とを単に関連づけ、または結合して、送受信したり、記憶装置に格納したりするようにしても良い。以上のように、本実施の形態は、ＸＭＬ文書２０１における構造２０２の部分を本実施の形態による手法を用いて圧縮し、さらに符号化された構造２０２の部分とコンテンツ２０４の部分とを、従来の手法を用いて圧縮する。このため、本実施の形態によるデータ圧縮は、全体としては、従来の種々の圧縮手法と併用して行うこととなる。

【００２８】以上のようにして圧縮されたＸＭＬ文書２０５を解凍する場合は、上記の圧縮課程の反対の過程をたどる。すなわち、まず解凍器２１において、圧縮器１３が圧縮符号化に用いた手法に対応する手法で、符号化データ列２０３とコンテンツ２０４とを解凍する。次に、デコーダ２３において、解凍された符号化データ列２０３を、文法規則１２と同一の文法規則２２を用いて複合化する。復号化処理の詳細な内容については後述する。文法規則２２はＤＴＤにより規定される。そして、この復号化処理により得られた構造２０２と解凍器２１により解凍されたコンテンツ２０４とを用いて、ＸＭＬ文書２０１を復元する。

【００２９】次に、本実施の形態によるＸＭＬ文書の構造に対する符号化処理の内容を詳細に説明する。ここでは、処理対象のＸＭＬ文書は、簡単のため、属性を含まないものとし、ＸＭＬ文書中のすべての実体は展開されているものとする。属性の扱いについては後述する。

【００３０】図３は、ＸＭＬ文書の構造を符号化するエンコーダ１１の構成を説明する機能ブロック図である。図３を参照すると、エンコーダ１１は、処理対象であるＸＭＬ文書２０１を構造２０２とコンテンツ２０４とに分割する分割処理部１１１と、文法規則１２に基づいて後述するプッシュダウンオートマトンを構築するオートマトン構築部１１２と、オートマトン構築部１１２により構築されたプッシュダウンオートマトンを符号化トランスデューサとして用いて構造２０２の符号化データ列２０３を生成する符号化データ列生成部１１３とを備える。

【００３１】図４に処理対象とするＸＭＬ文書の例を示す。ＸＭＬ文書のコンテンツとは、内容モデルにおける、＃ＰＣＤＡＴＡに相当する部分の文字列のリストである。すなわち、図４のＸＭＬ文書におけるコンテンツ

は、“String1”、“String2”、“String3”、“String4”という4つの文字列からなるリストである。これは、例えば、ナル文字で終わる文字列を並べたバイト列として、以下のように、コンパクトに表現可能である(但し“¥0”はナル文字をあらわす)。“String1¥0String2¥0String3¥0String4¥0”この文字列は、上述したように、構造部分とは別に圧縮符号化される。また、図４のＸＭＬ文書における構造は、図５に示すようになる。これは、図４のＸＭＬ文書中のコンテンツに相当する文字列をプレースホルダ（□）に置き換えたものである。

【００３２】本実施の形態において、エンコーダ１１は、分割処理部１１１により図４に示すＸＭＬ文書から図５に示す構造を取り出し、オートマトン構築部１１２により文法規則１２を用いてプッシュダウンオートマトンを構築し、符号化データ列生成部１１３により当該プッシュダウンオートマトンを用いて当該構造を符号化する。図６は、文法規則１２を規定するＤＴＤの例を示す。分割処理部１１１による分割処理の後、オートマトン構築部１１２は、文法規則１２を用いた符号化処理のために、当該ＤＴＤに対応するプッシュダウンオートマトンを構築する。図６のＤＴＤによれば、要素Ａが現れた場合は、次に、要素Ｂ、要素Ｃがこの順番で１回ずつ現れた後に終了することを示す。同様に、要素Ｂが現れたときは、次に、要素Ｄが１回現れた後に終了することを示す。また、要素Ｃが現れたときは、次に、要素Ｅまたは要素Ｆが０回以上現れた後に終了することを示す。さらにまた、要素Ｅが現れたときは、次に、要素Ｇまたは要素Ｈのいずれか一方が１回現れた後に終了することを示す。図７は、図６に示したＤＴＤに対応する自然なプッシュダウンオートマトンを示す図である。なお、非終端記号ＤとＧに関しては、終端記号＃ＰＣＤＡＴＡを取るだけの自明なルールなので、省略してある。このようなオートマトンは、文法の各非終端記号について曖昧さ無く構築できる。したがって、本実施の形態をデータ通信に用いる場合、送信側と受信側との共通のＤＴＤからは、全く同一のプッシュダウンオートマトンを構築することができる。

【００３３】通常、プッシュダウンオートマトンは、入力列の構文解析を行うために用いられる。その意味では、このプッシュダウンオートマトンは、表層のシンボル列、すなわち、＃ＰＣＤＡＴＡ（またはプレースホルダ“□”）の１個以上の並びからなる全ての列を受理する。しかし、解析の結果生成される構文解析木としては、例えば、構文木のノードＡの子供として、ノードＢ、ノードＣがこの順に現われなければならない。また、要素Ｃの後は、空遷移で最終状態に遷移する。このように、このプッシュダウンオートマトンは、解析済みのＸＭＬ文書（例えば、ＤＯＭ木）のような、構文解析木が文法を満たすかどうかのチェックに用いることもできる。

【００３４】図８に示す構文木を例として、プッシュダウンオートマトンを用いた文法のチェックについて説明する。なお、図８において、各リーフの＃ＰＣＤＡＴＡは省略してある。この構文木が図６のＤＴＤに規定される文法によって生成可能かどうかを調べるには、この構文木の各ノードに対して、そのノードの非終端記号に対応するオートマトンによって、その子ノードの列が受理できるかどうかを調べればよい。例えば、要素Ａは、子ノードとしてＢＣという列を持っている。これは、非終端記号Ａに対応するオートマトンによって受理される（図７のＡ参照）。したがって、この部分については文法を満たすことがわかる。同様にして、全てのノードについて、対応するオートマトンを使ってPreorderでトラバース（深さ優先でたどる）すれば、文法のチェックは終了する。構文解析木に対して、プッシュダウンオートマトンをこのように使うことを、以下の説明では妥当性検証と呼ぶ。なお、以上の操作で用いる各非終端記号に対するオートマトンについては、終了状態へのε遷移を除き、決定性で最小のオートマトンであるものとする。

【００３５】次に、オートマトン構築部１１２は、図７のプッシュダウンオートマトンを、ＸＭＬ文書の構造部分（図５参照）を符号化するトランスデューサ、すなわち、文字列の構文解析のオートマトンに変換する。図７のプッシュダウンオートマトンにおいて、入力を、４つの＃ＰＣＤＡＴＡ（またはプレースホルダー“□”）からなる並びとし、開始記号をＡとして解析を開始すると、ノードＡ、ノードＢ、ノードＤが順に作られ、最初の＃ＰＣＤＡＴＡが認識される。次に、ノードＣが作られたところで選択肢が発生する。すなわち、ノードＥを作るべきか、ノードＦを作るべきか、それともノードＣをこれで終わりにして上位のノードに戻るか、の３通りである。そこで、これらの３通りの選択肢に、ラベルのアルファベット順に１、２、３という番号を割り振る（ラベルεは常に最後と決める）。同様に、ルールＥの最初の状態も、ノードＧを作るかまたはノードＨを作るという選択肢を持つので、これらについても、１、２という番号を割り当てる。なお、ここでは選択肢に番号を割り振るとしたが、選択肢の識別に用いる符号は数字に限らない。アルファベットや記号など任意の符号を用いて選択肢を特定することが可能である。図９は、図７のプッシュダウンオートマトンから変換された符号化トランスデューサである。

【００３６】エンコーダ１１の符号化データ列生成部１１３は、オートマトン構築部１１２により構築された符号化トランスデューサを走らせる。図９に示す符号化トランスデューサは、妥当性検証（Preorderでのルール適用）を行った場合に、対応する選択肢番号があれば、その番号を出力する。すなわち、図９において、ルールＡ、Ｂ、Ｆ、Ｈに関しては、選択肢がないので何ら出力を行わず、ルールＣ及びルールＤが適用された場合は、

該当する番号を出力する。例えば、図８の構文木に対して妥当性検証を行った場合、木をたどるにしたがって、図１０に示すような番号を出力する。以上の処理により、“１１２１２.3”という番号列が得られる。この番号列は、プッシュダウンオートマトンの動きを厳密に規定している。したがって、この番号列を、図４に示したＸＭＬ文書の構造部分（図５）の符号化されたデータ列として扱うことができる。

【００３７】次に、本実施の形態によるＸＭＬ文書の構造に対する復号化処理の内容を説明する。上記の手順を経て符号化されたＸＭＬ文書を復号するには、符号化トランスデューサの入出力を逆にして適用すればよい。したがって、デコーダ２３は、図７と同一のプッシュダウンオートマトンを用いて復号化トランスデューサを生成し、復号化処理を実行する。上述したように、オートマトンは、文法の各非終端記号について曖昧さ無く構築できるため、ＤＴＤにて規定される文法規則１２と文法規則２２とが共通であれば、デコーダ２３においても図７と全く同一のプッシュダウンオートマトンを構築することができる。

【００３８】図１１は、図７と同一のプッシュダウンオートマトンから変換された復号化トランスデューサである。図１１に示す復号化トランスデューサにおいて、ｉ／Ｂという表現は、「ｉという入力文字列を見たら、Ｂというルールを呼び、その後、次の状態へ遷移する」という遷移をあらわす。これにより、エンコーダ１１から出力された番号列を入力して、対応する構文解析木を生成する。上述した“１１２１２３”という番号列を入力した場合、元々の番号の割り当て方から、このプッシュダウンオートマトン（復号化トランスデューサ）は、曖昧さ無くＸＭＬ文書の符号化番号列を受理することができる。したがって、生成される構文解析木は、図８に示したオリジナルの構文解析木と同一となる。これにより、当該ＸＭＬ文書の構造部分が復元されることとなる。

【００３９】次に、属性の扱いについて説明する。本実施の形態では、属性は、プッシュダウンオートマトンで扱えるように、木構造に変換する。具体的には、属性を持つ全ての要素（ELEMEMT）に関して、それらの属性を子ノードとして取るように変換する。この際、属性は属性名のアルファベット順に現れるものとする。そして、＃ＲＥＱＵＩＲＥＤである属性は、そのまま並べる。また、＃ＩＭＰＬＩＥＤである属性は、オプショナルである“？”をつける。なお、＃ＦＩＸＥＤである属性については、元々情報がないので、変換後のＤＴＤには含めない。図１２は、例示的に、所定のＤＴＤにおける変換前後の状態を比較する図である。図１２において、左側に示されるＤＴＤは、右側に示される形式に変換される。このようなＤＴＤに応じて、属性を含むＸＭＬ文書においても、属性を要素に変換する。図１３は、例示的に、所定のＸＭＬ文書における変換前後の状態を比較する図である。

【００４０】以上のようにして、ＤＴＤとＸＭＬ文書とを、属性を持たない状態とした後に、上述した符号化処理及び復号化処理を実行する。なお、ＤＴＤの変換は、プッシュダウンオートマトンを構築する前の段階で予め行っても良いし、プッシュダウンオートマトンを構築する段階で逐次行っても良い。前者の場合は、変換後の新しいＤＴＤを用いてプッシュダウンオートマトンを構築する手順となる。また、後者の場合は、元の（属性を持つ）ＤＴＤから直接プッシュダウンオートマトンを構成する手順となる。

【００４１】以上説明したように、本実施の形態は、ＸＭＬ文書を圧縮する側と解凍する側の双方において同一のＤＴＤを共有することが不可欠である。したがって、本実施の形態をデータ通信の際のデータ圧縮に用いる場合は、送信側データ処理装置と受信側データ処理装置とにそれぞれ同一のＤＴＤを用意する必要がある。図１４は、本実施の形態を用いたデータ通信システムの構成例を説明する図である。送信側データ処理装置１４１０において、エンコーダ１１は、データ処理部からＸＭＬ文書を受け取り、ＤＴＤ１４１１（図１の文法規則１２に相当）を用いて構造部分の符号化を行う。圧縮器１３は、符号化された構造部分及びコンテンツ部分の圧縮を行う。送信部１４１２は、エンコーダ１１及び圧縮器１３にて圧縮されたＸＭＬ文書を、通信ネットワークを介して受信側データ処理装置１４２０へ送信する。受信側データ処理装置１４２０において、受信部１４２２は通信ネットワークを介して受信した受信データを解凍器２１へ送る。解凍器２１は、受け取った受信データの解凍を行う。この時点でＸＭＬ文書のコンテンツ部分は復元される。デコーダ２３は、解凍された受信データの構造部分の符号化データ列を、ＤＴＤ１４２１（図１の文法規則２２に相当）を用いて復号化する。そして、解凍されているコンテンツ部分と合わせてＸＭＬ文書を復元し、データ処理部へ送る。ここで、送信側データ処理装置１４１０のＤＴＤ１４１１と受信側データ処理装置１４２０のＤＴＤ１４２１とが同一の内容となっている。なお、電子商取引等のビジネス間通信では、アプリケーションどうしでＸＭＬ文書をやりとりする場合、ＤＴＤは事前に合意されている場合がほとんどである。したがって、ＤＴＤを共有していることを前提に、本実施の形態をビジネス間通信に利用することができる。

【００４２】また、データベースシステムにおいて格納するデータファイルを圧縮するために用いる場合は、ＸＭＬファイルの構造を符号化するために用いたＤＴＤを、そのまま復号化する際に利用できるため、ＤＴＤが共有されているかどうかを考慮する必要はない。図１５は、本実施の形態を用いたデータベースシステムの構成を説明する図である。データベースシステム１５００に

15

おいて、エンコーダ１１は、データ処理部からＸＭＬ文書を受け取り、ＤＴＤ１５０１（図１の文法規則１２に相当）を用いて構造部分の符号化を行う。圧縮器１３は、符号化された構造部分及びコンテンツ部分の圧縮を行う。そして、エンコーダ１１及び圧縮器１３にて圧縮されたＸＭＬ文書が記憶装置１５０２に格納される。記憶装置１５０２に格納されている圧縮されたＸＭＬ文書を読み出す場合は、圧縮器１３が解凍器２１として動作し、エンコーダ１１がデコーダ２３として動作する。ＸＭＬ文書の構造部分の復号化には、符号化の際に用いたＤＴＤ１５０１を再度使用する。

【００４３】なお、以上の説明では、木ローカル言語としてＸＭＬを用いた場合を例として説明したが、ＡＳＮ．１等の他の木ローカル言語においてもそのまま利用することができる。ただし、この場合においても、データファイルを圧縮する側と解凍する側とで、上述したＸＭＬにおけるＤＴＤのような文法規則を共有することが必要である。

【００４４】

【発明の効果】以上説明したように、本発明によれば、木ローカル言語における文書の構造部分の符号化を行うデータ圧縮を実現することができる。

【００４５】また、木ローカル言語に特化すると共に、他の汎用的なデータ圧縮技術と併用することにより、高い圧縮率を期待できるデータ圧縮方法を提供することができる。

【図面の簡単な説明】

【図１】　本実施の形態における文書圧縮システムの全体構成を説明する図である。

【図２】　本実施の形態によるデータ圧縮の手順を説明する図である。

【図３】　本実施の形態におけるエンコーダの構成を説明する図である。

16

【図４】　本実施の形態の処理対象であるＸＭＬ文書の例を示す図である。

【図５】　図４のＸＭＬ文書における構造を示す図である。

【図６】　本実施の形態にて用いる文法規則の例を示す図である。

【図７】　図６の文法規則から構築されるプッシュダウンオートマトンを示す図である。

【図８】　プッシュダウンオートマトンを用いた文法のチェックの手法を説明するための構文木を例示的に示す図である。

【図９】　図７のプッシュダウンオートマトンを用いて生成された符号化トランスデューサを示す図である。

【図１０】　図８の構文木に対して妥当性検証を行った結果を例示的に示す図である。

【図１１】　図７と同一のプッシュダウンオートマトンを用いて生成された復号化トランスデューサを示す図である。

【図１２】　属性を持つＤＴＤを、属性を持たないＤＴＤに変換した状態を説明する図である。

【図１３】　属性を持つＸＭＬ文書を、属性を持たないＸＭＬ文書に変換した状態を説明する図である。

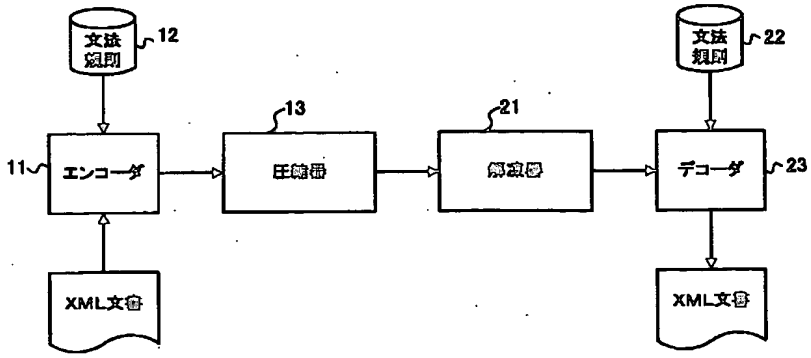
【図１４】　本実施の形態をデータ通信システムに適用した場合の構成を説明する図である。

【図１５】　本実施の形態をデータベースシステムに適用した場合の構成を説明する図である。

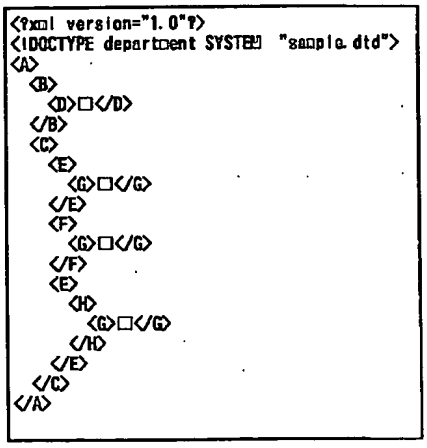
【符号の説明】

１１…エンコーダ、１２、２２…文法規則、１３…圧縮器、２１…解凍器、２３…デコーダ、２０１…ＸＭＬ文書、２０２…構造、２０３…符号化データ列、２０４…コンテンツ、２０５…圧縮されたＸＭＬ文書

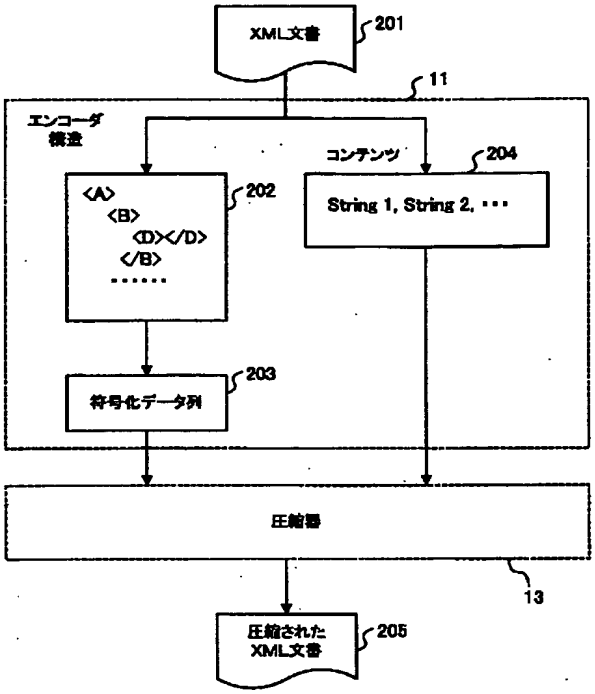
【図１】



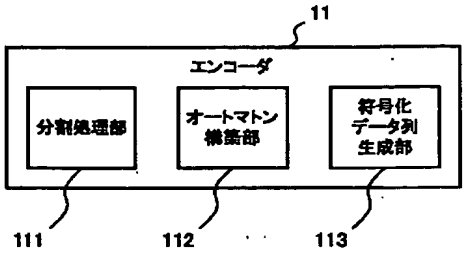
【図５】



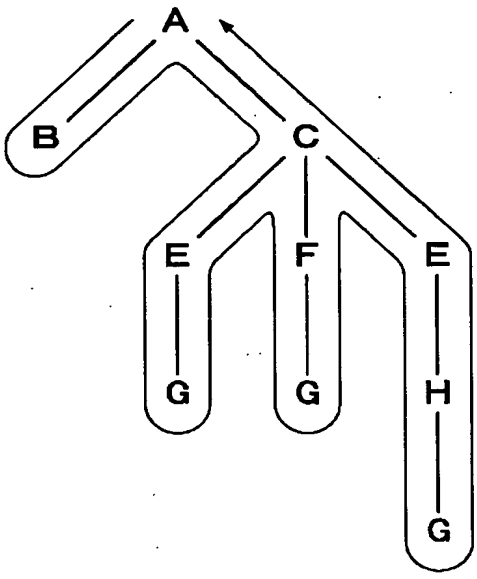
【図２】



【図３】



【図８】



【図４】

```
<?xml version="1.0"?>
<!DOCTYPE department SYSTEM "sample.dtd">
<A>
  <B>
    <D>String1</D>
  </B>
  <C>
    <E>
      <G>String2</G>
    </E>
    <F>
      <G>String3</G>
    </F>
    <E>
      <H>
        <G>String4</G>
      </H>
    </E>
  </C>
</A>
```

【図６】

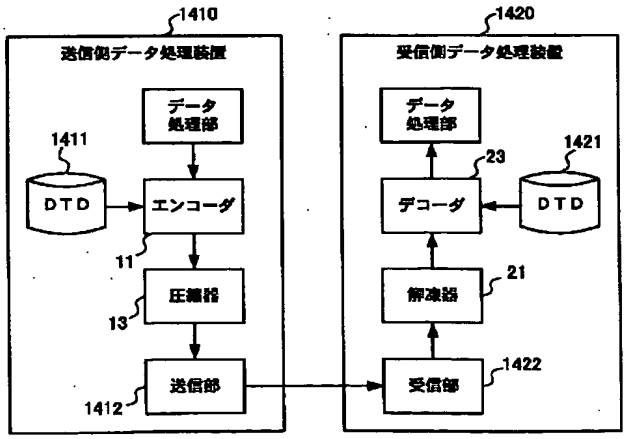
```
<ELEMENT A (B, C)>
<ELEMENT B (D)>
<ELEMENT C (E|F)*>
<ELEMENT E (G|H)>
<ELEMENT F (G)>
<ELEMENT H (G)>
<ELEMENT D (#PCDATA)>
<ELEMENT G (#PCDATA)>
```

【図１３】

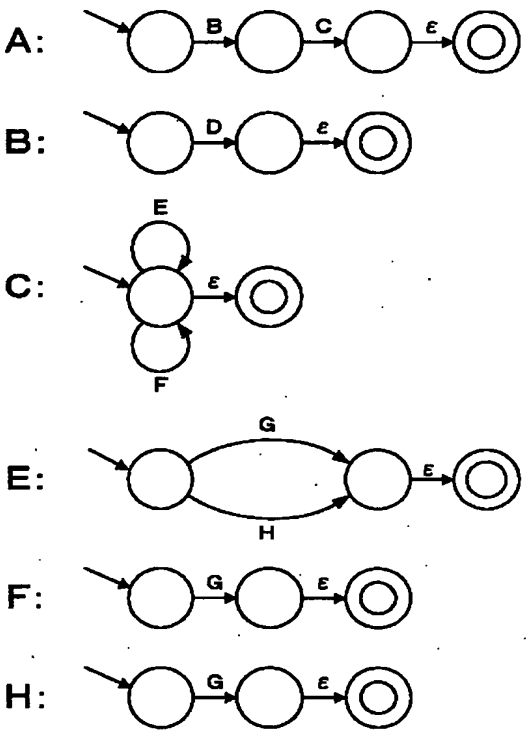
XML文書

変換前	変換後
<department serialNo="012345" manager="yes"> <name>aaaaa</name> <email>mail@address.com</email> </department>	<department <serialNo>012345</serialNo> <manager>yes</manager> <name>aaaaa</name> <email>mail@address.com</email> </department>

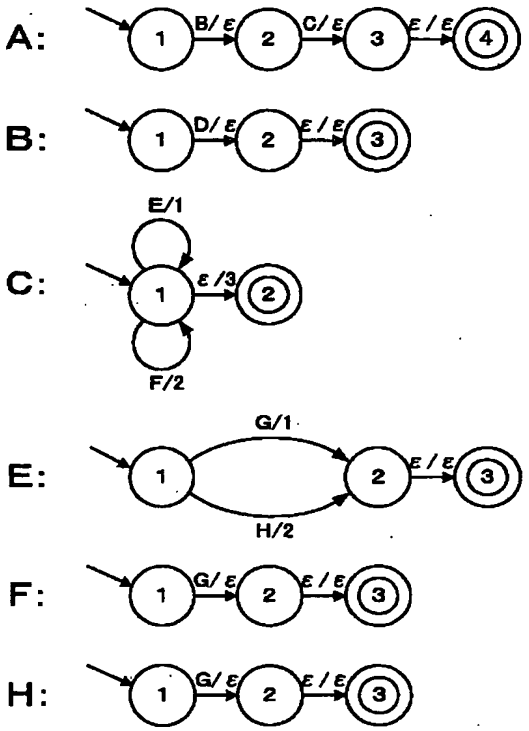
【図１４】



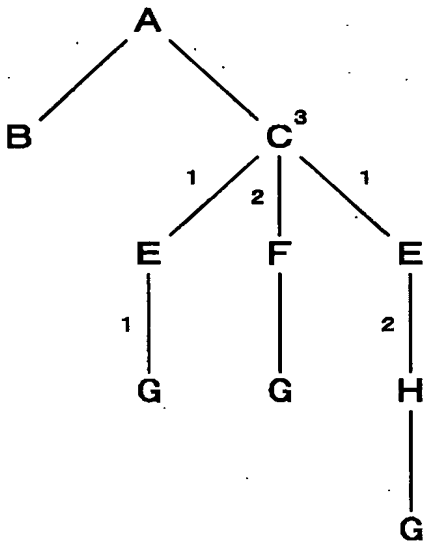
【図７】


A:
B
C
ε
B:
D
ε
C:
E
ε
F
E:
G
ε
H
F:
G
ε
H:
G
ε

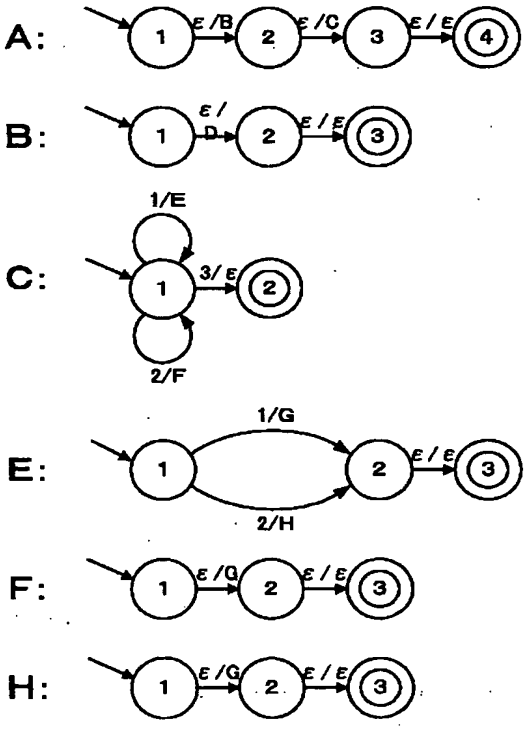
【図９】


A:
1
B/ε
2
C/ε
3
ε/ε
4
B:
1
D/ε
2
ε/ε
3
C:
E/1
1
ε/3
2
F/2
E:
G/1
1
2
ε/ε
3
H/2
F:
1
G/ε
2
ε/ε
3
H:
1
G/ε
2
ε/ε
3

【図１０】


A
B
C
3
1
2
1
E
F
E
1
2
G
G
H
G

【図１１】

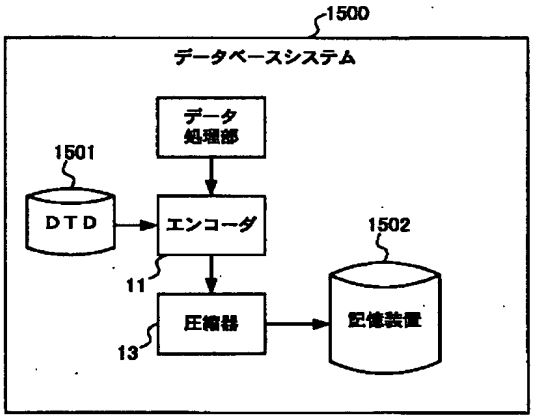

A:
1
ε/B
2
ε/C
3
ε/ε
4
B:
1
ε/D
2
ε/ε
3
C:
1/E
1
3/ε
2
2/F
E:
1/G
1
2
ε/ε
3
2/H
F:
1
ε/G
2
ε/ε
3
H:
1
ε/G
2
ε/ε
3

【図１２】

DTD

変換前	変換後
<!ELEMENT department (employee)> <!ELEMENT employee (name, email)> <!ATTLIST employee serialNo CDATA #REQUIRED> <!ATTLIST employee manager CDATA #IMPLIED> <!ELEMENT name (#PCDATA)> <!ELEMENT email (#PCDATA)>	<!ELEMENT department (employee)> <!ELEMENT employee (serialNo, manager?, name, email)> <!ELEMENT serialNo (#PCDATA)> <!ELEMENT manager (#PCDATA)> <!ELEMENT name (#PCDATA)> <!ELEMENT email (#PCDATA)>

【図１５】



フロントページの続き

(72)発明者　丸山　宏
神奈川県大和市下鶴間1623番地14　日本アイ・ビー・エム株式会社　東京基礎研究所内

(72)発明者　田村　健人
神奈川県大和市下鶴間1623番地14　日本アイ・ビー・エム株式会社　東京基礎研究所内

(72)発明者　浦本　直彦
神奈川県大和市下鶴間1623番地14　日本アイ・ビー・エム株式会社　東京基礎研究所内

Ｆターム(参考)　5J064 AA00 BA11 BA14 BC02 BD02 BD03